

WHAT IS CLAIMED IS:

1. A system for testing the implementation of a software system for performing a required business process comprising:
  - (a) a business process specification for specifying the business process in a BPML compliant language;
  - (b) an analysis module for generating a plurality of possible valid test scenarios;
  - (c) a directed random engine or a constraint solving system for randomly generating tests according to said test scenarios, such that said tests are limited according to at least one of a business priority and a testing priority; and
  - (d) a simulator for determining expected results for said generated tests.
2. The system of claim 1, wherein said simulator determines said expected results according to said business process specification
3. The system of claim 1, wherein the software system is determined according to a software specification, said software specification determining inputs for said generating said tests by said directed random engine or said constraint solving system, said inputs also being determined according to said business process specification.
4. The system of claim 1, further comprising:

a modeling module for receiving said rules from said business process specification and applying said results from executing said generated tests.

5. The system of claim 4, wherein said modeling module models behavior of the business process and generates output representing predicted results for said simulator.

6. The claim of claim 5, further comprising: a validation module for comparing said expected results to said results from executing said generated tests.

7. The system of claim 6, further comprising:  
a coverage system for calculating coverage provided by said generated tests.

8. The system of claim 1, further comprising  
connector hub technology for translating said plurality of tests into concrete calls to the software system.

9. The system of claim 7, wherein said analysis module analyzes each transition of the business process according to said business process specification.

10. The system of claim 9, wherein said transition comprises a starting state, a target state, and a condition or event that causes a change from said starting state to said

target state.

11. The system of claim 10, wherein at least one test primitive is determined from at least one transition.

12. A method for testing the implementation of a software system for performing a required business process, comprising:  
providing a specification for describing the business process;  
analyzing said specification to form an analysis of the business process; and  
generating at least one test for testing the software system for performing the business process according to said analysis.

13. The method of claim 12, wherein said specification comprises at least one general requirement for performing the business process and at least one rule for being fulfilled by the software system.

14. The method of claim 12, further comprising:  
performing said at least one test to obtain a result; and  
analyzing said result to determine a performance of the software system.

15. The method of claim 14, wherein analyzing said result further comprises determining coverage provided by said at least one test.

16. The method of claim 15, wherein said generating said at least one test further comprises:

- performing an initial generation of at least one test;
- performing said at least one test to obtain a result;
- analyzing said result to determine coverage of said at least one test; and
- generating a plurality of tests according to said coverage.

17. The method of claim 16, wherein said analyzing said result to determine said performance of the software system further comprises:

- determining an expected result from the software system according to said specification; and
- comparing said expected result with an actual result to determine said performance of the software system.

18. The method of claim 12, wherein said generating is performed at least partially according to a directed random generation engine.

19. The method of claim 12, wherein the software system implements a plurality of business processes as a business application.

20. The method of claim 19, wherein said business application is selected from the group consisting of billing, Enterprise Resource Planning (ERP), Customer Requirements Management (CRM), Supply Chain Management (SCM), Human

Resource management.

21. A system for automatic verification of the implementation of a software system for performing a required business process, the business process being described according to a specification, the system comprising:

- (a) a generator for automatically generating tests from at least one rule specified in the specification of the business process;
- (b) a simulator for generating at least one expected result of said tests from the specification;
- (c) a connector for receiving an actual result of said tests from the software system being tested; and
- (d) a validator for comparing said actual result with said at least one expected result.

22. The system of claim 21 further comprising:

- (e) a data entry system for entering said at least one specification of at least one process.

23. The system of claim 21, wherein said connector receives said actual result of said tests from a simulation of said software system.

24. The system of claim 21, wherein said generator comprises a directed random generation engine.

25. A method for verification of a software system for performing a business process comprising:

- modeling the business process to form a model;
- analyzing the model according to a plurality of actions occurring in the model;
- developing at least one test strategy according to said plurality of actions; and
- generating at least one test according to said at least one test strategy.

26. The method of claim 25, wherein developing said at least one test strategy, further comprises:

- determining priority with respect to said test;
- controlling and optimizing for corner cases and risk points.

27. The method of claim 26 wherein generating said at least one test further comprises:

- controlling test runs; and
- determining an analysis, comparison and coverage of test results from said test runs.

28. The method of claim 26 wherein said generating further comprises:

- generating scripts; and
- connecting to a connector for operating the test on the software system.

29. A method for testing web services based implementation of a software system, the software system performing a process, comprising: providing a specification for describing the process and the Web services;

analyzing said specification to form an analysis of the process and the Web services; and

generating at least one test for testing the software system for performing the process according to said analysis.

30. The method of claim 29, wherein the Web services perform, integrate with or connect to, a business application selected from the group consisting of billing, Enterprise Resource Planning (ERP), Customer Requirements Management (CRM), Supply Chain Management (SCM), Human Resource management.

31. The method of claim 30, wherein said business application is described in a formal language, selected from the group consisting of UML (unified modeling language) activity diagrams, UML sequence diagrams or UML state charts, BPEL (business process execution language) standard language, BPML (business process modeling language) standard language, or any other equivalent language.

32. A method for testing software for performing a business process, comprising:

analyzing the business process according to a plurality of general requirements and rules for the business process;

analyzing a specific implementation of the business process as software;  
generating at least one abstract test for examining a behavior of the software as  
said specific implementation of the business process.

33. The method of claim 32, further comprising:  
generating at least one detailed test description according to a plurality of  
constraints, said constraints being determined according to operating parameters of the  
software; and  
generating at least one script according to said at least one detailed test  
description.

34. The method of claim 33, further comprising:  
executing said at least one script;  
analyzing results of said executing to assess performance of the software.

35. The method of claim 32, wherein said analyzing said specific  
implementation of the business process comprises determining a software specification  
for specifying a plurality of functions for the software.

36. The method of claim 35, further comprising:  
assessing conformance of the software according to said software specification.

37. The method of claim 32, wherein said analyzing the business process



further comprises:

determining a business process description.

38. The method of claim 37, wherein said business process description is provided in a formal language.

39. The method of claim 38, wherein said formal language is selected from the group consisting of UML (unified modeling language) activity diagrams, UML sequence diagrams, UML state charts, BPEL (business process execution language) standard language, or BPML (business process modeling language) standard language.

40. The method of claim 38, wherein analyzing the business process further comprises parsing said business process description.

41. The method of claim 37, wherein said analyzing the business process further comprises:

analyzing each transition of the business process.

42. The method of claim 41, wherein said transition comprises a starting state, a target state, and a condition or event that causes a change from said starting state to said target state.

43. The method of claim 41, wherein at least one test primitive for at least partially determining at least one abstract test is determined from at least one transition.

44. The method of claim 43, wherein a plurality of test primitives is determined for constructing a tree for determining said at least one abstract test.

45. The method of claim 43, wherein said at least one detailed test description is at least partially determined from at least one input and at least one event for said at least one transition.

46. The method of claim 44, wherein a plurality of tests are generated from said at least one detailed test description according to a directed random generator.

47. The method of claim 46, further comprising:  
determining at least potential functional coverage of said plurality of tests by comparison to a set of all possible tests.

48. The method of claim 32, further comprising:  
translating said plurality of tests into concrete calls to the tested software by using a connector hub technology.

49. The method of claim 48, further comprising:

generating a plurality of inputs according to the business process description; and  
executing said plurality of tests.

50. The method of claim 49, further comprising:  
analyzing results from said executing said plurality of tests for comparing with a  
plurality of expected results, said expected results being determined according to said  
business process description.

51. The method of claim 32, wherein the business process comprises a process  
selected from group consisting of billing, marketing and distribution, and personnel  
management.

52. The method of claim 32, wherein said analyzing said specific  
implementation of the business process comprises generating a finite state machine  
description of said plurality of general requirements and rules.